

Institute for Bioinformatics and Medical Informatics (IBMI)

Simulating Biological Systems with OptSolvX-powered SBSCl

Michael Gaas^{1,2†}, Ayush Baranwal^{3†}, Arthur Neumann^{1,2}, Matthias König^{4,5}, and Andreas Dräger⁶

¹German Center for Infection Research (DZIF), partner site Tübingen, Tübingen 72076, DE, ²Quantitative Biology Center (QBiC), Eberhard Karl University of Tübingen, Tübingen 72076, DE, ³School of Engineering, Jawaharlal Nehru University, New Dehli, IN, ⁴University Hospital Schleswig-Holstein, Campus Lübeck, First Department of Medicine, 23538 Lübeck, DE, ⁵Humboldt-Universität zu Berlin, Faculty of Life Sciences, Department of Biology, Institute of Theoretical Biology, Systems Medicine of the Liver, Unter den Linden 6, 10099 Berlin, DE, ⁶Data Analytics and Bioinformatics, Institute of Computer Science, Martin Luther University Halle-Wittenberg, Halle (Saale) 06120, DE, [†]Authors contributed equally

Background

Studying biological systems increasingly relies on large numerical computations and complex simulation workflows. The Systems Biology Simulation Core Library (SBSCl) is a Java-based software library for simulating models and experiments encoded in standard formats such as SBML (Systems Biology Markup Language) and SED-ML (Simulation Experiment Description Markup Language).

When simulating complex models, time-varying stiffness poses a significant numerical challenge. SBSCl's Java™ implementation of LSODA (Livermore Solver for Ordinary Differential Equations with Automatic Method Switching) addresses this by automatically switching between suitable ODE integration methods during simulation, decreasing redundant computational cost.

For constraint-based simulations, such as Flux Balance Analysis (FBA), solver integration has traditionally depended on specific linear programming (LP) backends. OptSolvX addresses this limitation by adding a solver-agnostic LP core to SBSCl, separating biological model interpretation from solver-specific implementation details.

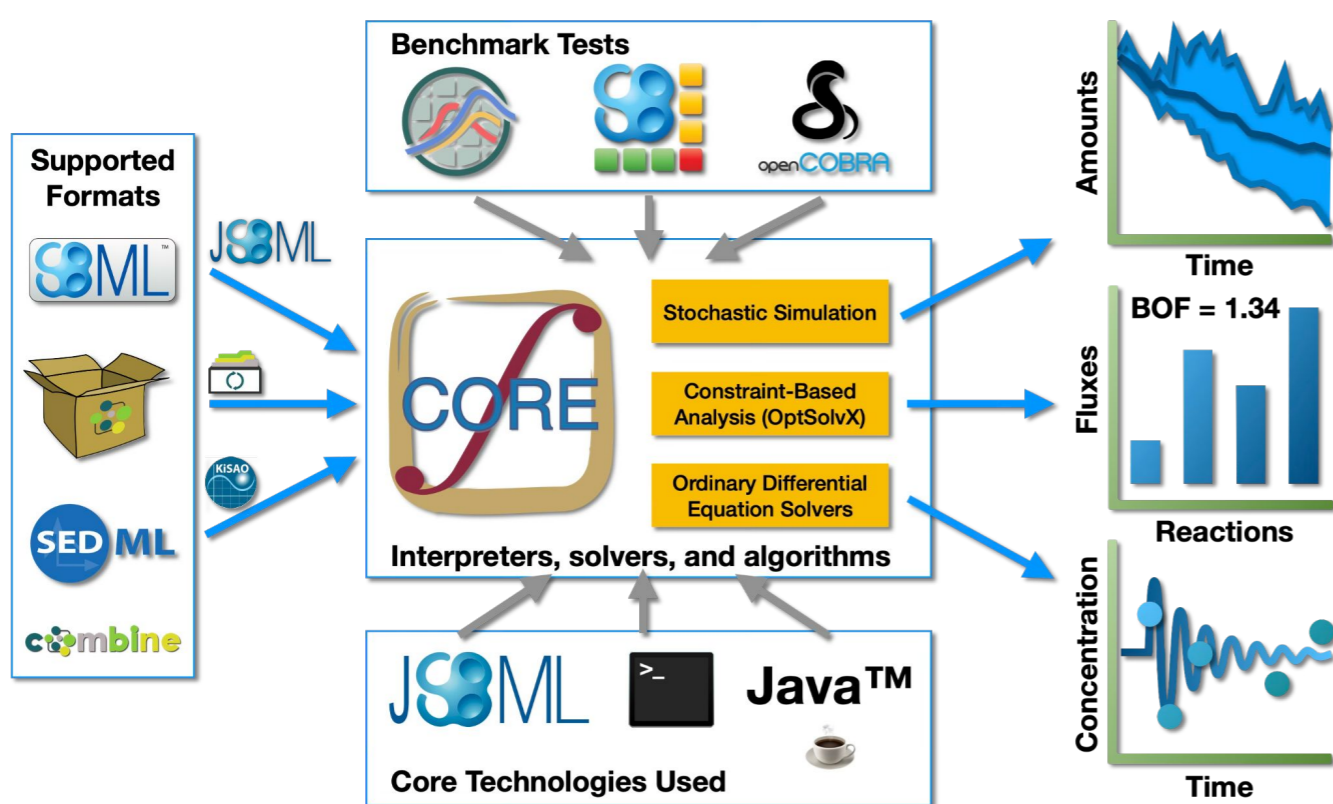


Fig. 1 | Constraint-based simulation workflow with SBSCl and OptSolvX. SBSCl parses standardized systems biology models and delegates constraint-based simulation tasks to OptSolvX. The bridge translates SBML/FBC information such as flux bounds, objective definitions, and steady-state constraints into a solver-independent linear programming model. OptSolvX then forwards the model to an interchangeable solver backend and returns the computed flux distribution.

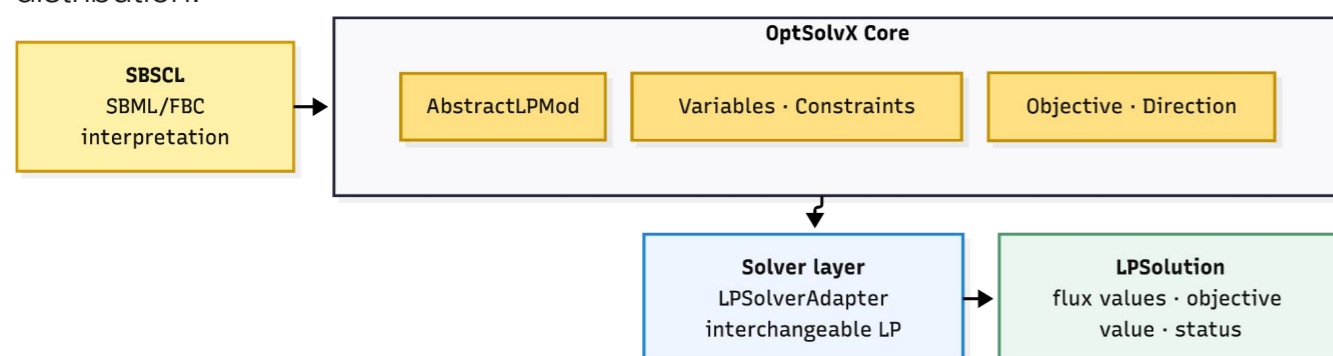


Fig. 2 | OptSolvX architecture for solver-agnostic constraint-based simulation. SBSCl translates SBML/FBC model information into an OptSolvX linear programming model. The OptSolvX core stores variables, constraints, bounds, objective function, and optimization direction independently of any concrete backend. The solver layer connects this model to interchangeable LP solvers via a common adapter interface and returns the computed flux distribution, the objective value and the solution status.

University of Tübingen
Department of Computer Science
Sand 14 · 72076 Tübingen · Germany



Results and Improvements

When a system's stiffness changes, expensive stiff ODE methods become computationally infeasible. LSODA solves this by automatically detecting change in stiffness and switching between Adams method for non-stiff and BDF for stiff systems.

Methods like FBA strongly rely on LP solver backends. Different LP solvers vary significantly in performance, licensing and availability across platforms. OptSolvX adds a backend-independent optimization layer to SBSCl. Models are represented as variables, constraints and objectives and solved through interchangeable LP solver adapters.

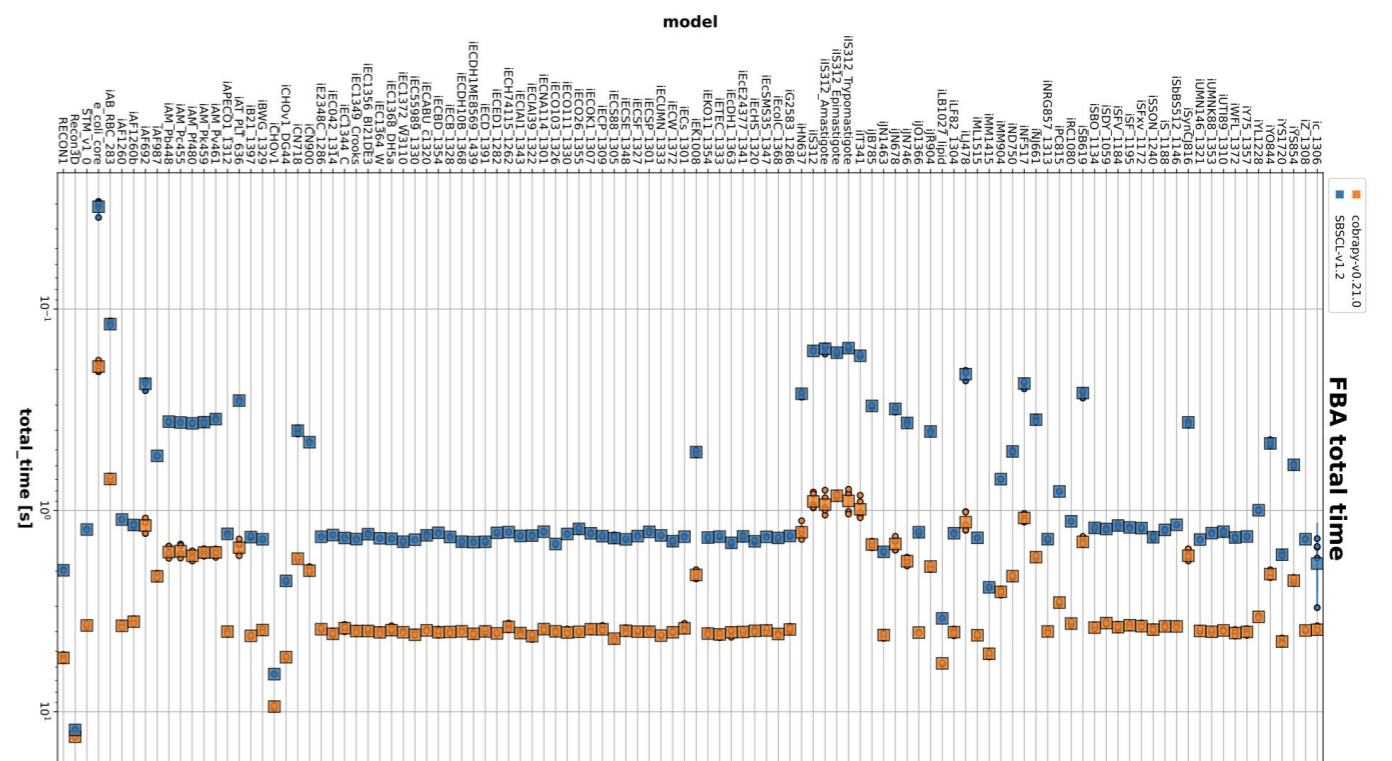


Fig. 3 | FBA total time on BiGG models. The combined model loading and Flux Balance Analysis simulation time for each model from the BiGG Models database. SBSCl is compared with COBRApy, with mean values, standard deviations, and individual repeats shown for five runs per model on a logarithmic time scale.

Discussion

LSODA extends SBSCl's kinetic simulation support by automatically switching between non-stiff and stiff ODE integration methods when model stiffness changes. However, for models that remain non-stiff, stiffness detection may introduce additional overhead, making LSODA most useful for simulations with changing numerical stiffness.

For constraint-based simulations, the OptSolvX integration turns solver choice into an implementation detail of SBSCl. This makes FBA easier to maintain, test and extend while preserving the user-facing workflow for biological models.

Availability

SBSCl and OptSolvX are freely available at <https://draeger-lab.github.io/SBSCl/> and at <https://github.com/draeger-lab/OptSolvX>. For more information, see the publication at doi: [10.1093/bioinformatics/btab669](https://doi.org/10.1093/bioinformatics/btab669)

References

- Panchiwala et al. 2022. doi.org/10.1093/bioinformatics/btab669
- Keller et al. 2013. doi.org/10.1186/1752-0509-7-55
- Keating et al. 2020. doi.org/10.15252/msb.20199110
- Keller et al. 2013. doi.org/10.1186/1752-0509-7-55
- Norsigian et al. 2020. doi.org/10.1093/nar/gkz1054
- Rodriguez et al. 2015. doi.org/10.1093/bioinformatics/btv341
- Waltemath et al. 2011. doi.org/10.1186/1752-0509-5-198

[linkedin.com/in/neumannarthur](https://www.linkedin.com/in/neumannarthur)
[linkedin.com/in/gaasmichael](https://www.linkedin.com/in/gaasmichael)

